Amendment to the Claims:

This listing of claims replaces all prior versions, and listings, of claims in the application:

1. (Currently Amended) A method of inter-thread communication in a multi-threaded computer comprises:

storing an inter-thread message in memory, the inter-thread message having a field for an address that indicates a location of data for a next thread to execute; and

writing to a self-destruct register after storing the message to indicate that a thread which stored the message in memory has completed execution, with the self-destruct register ~~being~~ configured to automatically clear ~~cleared~~ upon reading ~~by the next thread~~.

2. (original) The method of claim 1 wherein the inter-thread message field for an address provides an address of a register where the data for the next executing thread is stored.

3. (original) The method of claim 1 wherein writing to a self-destruct register further comprises:

setting at least one bit in the self-destruct register which corresponds to the thread which is writing to the self-destruct register.

4. (original) The method of claim 1 wherein writing to a self-destruct register further comprises:

setting one bit in the self-destruct register which corresponds to the thread which is writing to the self-destruct register.

5. (original) The method of claim 1 further comprising:

writing to the self-destruct register by a first thread;

reading from the self-destruct register by a second thread, where the reading further comprises:

reading bits, if any, that are set in the self-destruct register; and

clearing all of the bits of the self-destruct register.

6. (original) The method of claim 5 further comprises:

reading the inter-thread message from the memory by a new thread, where the new thread is a thread other than the first thread; and

executing the new thread.

7. (Currently Amended) A hardware-based multi-threaded processor comprises:

a general purpose processor ~~that coordinates~~ configured to coordinate system functions;

a plurality of microengines ~~that~~ configured to support multiple thread execution;

a scratchpad memory ~~for storing~~ configured to store inter-thread messages where execution of a write to the scratchpad memory by a first thread causes an address to be stored as an inter-thread message which indicates a location of data for a ~~new~~ different thread; and

a self-destruct register ~~for indicating~~ configured to indicate the execution status of threads where reading of the self-destruct register automatically clears all of the bits of the self-destruct register.

8. (original) The processor of claim 7 wherein the plurality of microengines further comprise:

a register stack wherein execution of the write to the scratchpad memory by the first thread causes a register address of the register referenced by the write to be stored as the inter-thread message, with the register address indicating the register stack location of data for the new thread.

9. (original) The processor of claim 7 wherein execution of a write to the self-destruct register by a thread causes at

least one bit to be set in the self-destruct register, the bit being set corresponding to the thread which executed the write to the self-destruct register.

10. (original) The processor of claim 7 wherein execution of a write to the self-destruct register by a thread causes one bit to be set in the self-destruct register, the bit set corresponding to the first thread which executed the write to the self-destruct register.

12-15. (Cancelled)

16. (new) The processor of claim 7 wherein the read from the self-destruct register by the thread causes execution of a new thread for each bit that is set, if any, in the self-destruct register.

17. (new) A computer program product residing on a computer readable medium, the computer program product including instructions to cause one or more machines to perform operations comprising:

storing an inter-thread message in memory; and

setting at least one bit in a self-destruct register in response to storing the inter-thread message in a memory, the

self-destruct register configured to automatically clear the at least one bit in response to a read of the at least one bit.

18. (new) The computer program of claim 17 wherein the operations further comprise:

storing a register address as the inter-thread message in memory.

19. (new) The computer program of claim 17 wherein the operations further comprise:

reading the contents of the self-destruct register, the reading automatically clearing the self-destruct register; and

executing a new thread according to any bits set in the self-destruct register.

20. (New) The method of claim 1, wherein the self destruct register includes a plurality of bits, and wherein, in response to a read of one or more of the plurality of bits, the self destruct register is configured to automatically clear the one or more of the plurality of bits.

21. (New) The method of claim 1, wherein each of the plurality of bits is associated with one of a plurality of threads.

22. (New) The processor of claim 7, wherein the self destruct register includes a plurality of bits, and wherein, in response to a read of one or more of the plurality of bits, the self destruct register is configured to automatically clear the one or more of the plurality of bits.

23. (New) The processor of claim 7, wherein each of the plurality of bits is associated with one of a plurality of threads.

24. (New) The computer program of claim 17, wherein the self destruct register includes a plurality of bits, and wherein, in response to a read of one or more of the plurality of bits, the self destruct register is configured to automatically clear the one or more of the plurality of bits.

26. (New) The computer program of claim 17, wherein each of the plurality of bits is associated with one of a plurality of threads.